IN THE CLAIMS

Please amend the claims as follows:

1.   (Cancelled)

2.   (Previously Presented) A method according to claim 11, wherein the step of synchronizing includes the step of:

    blocking a commit of the transaction until the number of transactions in the buffers is in a predetermined numerical relationship with the predetermined number of transactions.

3.   (Previously Presented) A method according to claim 2, wherein:

    said blocking the commit of the transaction until the number of transactions in the buffers is in the predetermined numerical relationship with the predetermined number of transactions including blocking the commit of the transaction until the number of transactions in the buffers is less than the predetermined number of transactions.

4.   (Cancelled)

5.   (Previously Presented) A method according to claim 11, wherein:

    the log writer process performs the step of synchronizing.

6.   (Previously Presented) A method according to claim 11, wherein:

    a database application process performs the step of synchronizing before submitting the transaction to the log writer process.

7.   (Cancelled)

8.   (Previously Presented) A method according to claim 11, further comprising the steps of:

    receiving input from an operator indicating a transaction loss bound; and

    setting the predetermined number of transactions based on the transaction loss bound.

2

9.   (Cancelled)

10.   (Currently Amended) A computer-readable medium bearing instructions for causing one or more processors to performs the steps of the method according to claim 11.

11.   (Original) A method for replicating data of a primary database system, comprising the steps of:

       maintaining a queue of transactions to be sent to a standby database system;

       storing a counter indicating a number of the transactions in the queue;

       storing a predetermined bound of transactions;

       executing a log writer process to:

              record the transaction in a redo log,

              compare the counter and the predetermined bound,

              if the counter is not less than the predetermined bound, then block a commit of the transactions until the counter is less than the predetermined bound, and

              if the counter is less than the predetermined bound, then increment the counter and acknowledge the commit of the transaction; and

       executing a net server process to:

              transmit the transaction over a network connection to the standby database system,

              receive an acknowledgement that a redo record for the transaction has been written to a standby log at the standby database system, and

              in response to the acknowledgement, remove the transaction from the queue and decrement the counter.

12-15. (Cancelled)

16.   (Previously Presented) A method for replicating data in a primary database system having multiple database servers operating in parallel and accessing a common database on a shared disk, said method comprising the steps of:

        setting a bound for each of the multiple database servers;

        for each of the multiple database servers, performing the steps of:

            maintaining a buffer of transactions to be sent to a standby database system; and

            synchronizing a transaction performed on the primary database system based on a number of transactions in the buffer and the corresponding bound;

        storing a counter indicating a number of the transactions in the queue;

        storing a predetermined bound of transactions;

        executing a log writer process to:

            record the transaction in a redo log,

            compare the counter and the predetermined bound,

            if the counter is not less than the predetermined bound, then block a commit of the transaction until the counter is less than the predetermined bound, and

            if the counter is less than the predetermined bound, then increment the counter and acknowledge the commit of the transaction; and

        executing a net server process to:

            transmit the transaction over a network connection to the standby database system, receive an acknowledgement that a redo record for the transaction has been written to a standby log at the standby database system, and

            in response to the acknowledgement, remove the transaction from the queue and decrement the counter.

17.   (Previously Presented) The method according to claim 16, wherein the step of synchronizing includes the step of blocking a commit of the transaction until the number of transactions in the buffers is in a predetermined numerical relationship with the predetermined number of transactions.

18.   (Previously Presented) The method according to claim 16, wherein said blocking the commit the transaction until the number of transactions in the buffer is in the predetermined

numerical relationship with the predetermined number of transaction including blocking the commit of the transaction until the number of transactions in the buffer is less than the predetermined number of transactions.

19. (Previously Presented) The method according to claim 16, wherein the log writer process performs the step of synchronizing.

20. (Previously Presented) The method according to claim 16, wherein a database application process performs the step of synchronizing before submitting the transaction to the log writer process.

[[20.]] 21. (Currently Amended) The method according to claim 16, further comprising the steps of:

      receiving input from an operator indicating a transaction loss bound; and

      setting the predetermined number of transactions based on the transaction loss bound.

[[21.]] 22. (Currently Amended) A computer-readable medium bearing instructions for causing one or more processors to perform the steps of the method according to claim 16.